IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of	:	JACK C. WYBENGA, et al
Serial No.	:	10/658,977
Filed	:	September 10, 2003
For	:	APPARATUS AND METHOD FOR PERFORMING HIGH-SPEED LOOKUPS IN A ROUTING TABLE
Group No.	:	2189
Examiner	:	Horace L. Flournoy

MAIL STOP AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request.

This request is being filed with a notice of appeal. The review is requested for the reason(s) stated in the arguments below, demonstrating the clear legal and factual deficiency of the rejections of some or all claims.

Claims 1-22 were rejected as anticipated by Lipman et al. (USP 6,192,051, hereinafter "Lipman"). These rejections are traversed. For the convenience of the Panel, claim 1 is reproduced below.

1. For use in a router, a lookup circuit for translating received addresses into destination addresses comprising:

M pipelined memory circuits for storing a trie table capable of translating a first received address into a first destination address,

wherein said M memory circuits are pipelined such that a first portion of said first received address accesses an address table in a first memory circuit and an output of said first memory circuit accesses an address table in a second memory circuit.

Independent claims 1 and 11 each require "a first portion of said first received address accesses an address table in a first memory circuit and an output of said first memory circuit accesses an address table in a second memory circuit". This feature is not taught or suggested by Lipman.

While Lipman's Figure 11 does show that IP address bits [31:16] provide the index of an entry in the level-1 tree 150 (col. 15, lines 59-61), the output "Next Tree Index" (the "NT pointer") is not used to "access[] an address table in a second memory circuit" where the second memory circuit is one of the "M pipelined memory circuits", as claimed in Claims 1 and 11. Instead, the NT pointer from the level-1 tree 150 is used as an index into the level-2 next tree table 152 of the forwarding table (col. 15, lines 65-67). That is, rather than being used to access the next memory circuit in a pipeline of memory circuits forming a trie table, it appears to be used for a separate lookup as an index in a forwarding table. The output of that forwarding table appears to be used as an offset index, added to T2 base, into the level-2 tree 154.

Figure 11 shows that the NT pointer from the level-1 tree 150 is used as an index into the level-2 next tree table 152 of the forwarding table (col. 15, lines 65-67), and nothing teaches or suggests that the forwarding table . That is, rather than being used to access the next memory circuit in a pipeline of memory circuits forming a trie table, it appears to be used for a separate lookup as an

index in a forwarding table. The output of that forwarding table appears to be used as an offset index, added to T2 base, into the level-2 tree 154.

The Examiner also indicates that Lipman's next tree table 152 is the claimed "address table in the second memory circuit". Of course, by combining element 144 from Figure 7 and element 152 of Figure 11, the Examiner is unable to show at all that there are any pipelined memory circuits, as these two figures do not show any common elements interrelating. Figure 11 is describing a compressed tree, Figure 7 is describing an uncompressed tree, and these do not interrelate.

The Examiner now responds with a broad interpretation of "memory circuit" that describes "a combination of interconnected electrical components or pathways that perform a specific task, that being data storage", and indicates that these are shown in Figure 8. Figure 8 shows a data structure, by Lipman's own description, not any interconnected electrical components or pathways corresponding to the Examiner's own definition. Figure 7 is also a data structure. Data structures are not memory circuits.

Certainly a data structure is stored in a physical memory comprising memory circuits. However, a typical physical memory is not comprised of pipelined memory circuits, and certainly not in the manner claimed. Nothing in Lipman describes pipelined memory circuits as claimed.

In fact, though Lipman shows multiple tables that point to each other, nothing in Lipman's description teaches or suggests a lookup circuit comprising "M pipelined memory circuits" as claimed. The Examiner's statement that "these memory circuits are 'pipelined' in that they each point to another circuit" is a factually incorrect statement. Those of skill in the art recognize that a "pipeline" is a series of elements each connected so that the output of one element is an input of the

next element. See, for example, Figure 3 of the present application, reproduced below, where the output of each pipelined SRAM is an input to the next one in series. Lipman does not appear to teach or suggest such a pipeline at all, much less pipelined memory circuits.

The Examiner refers to Lipman's Figure 7 for "pipelined memory circuits", but this figure shows a diagram of a data structure, not memory circuits. If the Examiner is intending to imply that the actual semiconductor circuits that comprise each bit/cell of a memory are the "memory circuits", then the Examiner will surely recognize that these are not typically pipelined, and Lipman certainly doesn't teach any such pipelining. Lipman doesn't teach series connections or pipelining of memory circuits at all, and the rejections are legally deficient.

The Examiner responds that "Lipman teaches pipelining via the levels of the 'tree' that are accessed in sequence. For example, a first tree level is accessed, followed by a second level, and third (i.e. a pipeline)." The Examiner's response illustrates his misunderstanding. The sort of sequential access described by the Examiner is not a pipeline, as recognized by those of skill in the art. In a pipeline, data passes from a first element, directly to the next element, and then directly to the element after that. This is very different from, for example, a processor that sends data to/from a first element, then sends data to/from a second element, then sends data to/from a third element. A pipeline of hardware circuits is connected in series, not merely accessed in sequence. The rejection is also factually deficient.

The Examiner's statement that "Pipelining is a term which is used to identify strings of several data or objects" is simply incorrect, particularly as applied to "strings of data". The Examiner was unable to provide any basis for this remarkable definition. In the Advisory Action, the

Examiner makes the statement that "Pointers are used as simply [*sic*] representations of elements within circuits that are themselves inputs to another element." Applicant respectfully notes that the Examiner may not fully understand the difference between a data structure and a physical hardware circuit – a pointer in a data structure normally has no relation at all to the underlying hardware, and nothing in Lipman suggests that it does so in this case.

Lipman fails to teach or suggest significant limitations of every independent claim, and all rejections are both legally and factually deficient.

CONCLUSION

As a result of the foregoing, the Applicant asserts that the claims in the Application are in condition for allowance over all art of record, and respectfully requests this case be returned to the Examiner for allowance or, alternatively, further examination.

The Commissioner is hereby authorized to charge any additional fees connected with this communication or credit any overpayment to Deposit Account No. 50-0208.

Respectfully submitted,

MUNCK BUTRUS CARTER P.C.

John T. Mockler

Date: February 19, 2008

John T. Mockler
Registration No. 39,775

P.O. Drawer 800889
Dallas, Texas 75380
(972) 628-3600 (main number)
(972) 628-3616 (fax)
E-mail: *jmockler@munckbutrus.com*